Deep Author Name Disambiguation using DBLP Data

Zeyd Boukhers[1,2*] and Nagaraj Bahubali Asundi[1]

[1]Institute for Web Science and Technologies (WeST), University of Koblenz-Landau, Universitätsstraße 1, Koblenz, 56070, Germany.
[2]Department of Data Science and Artificial Intelligence, Fraunhofer Institute for Applied Information Technology -FIT-, Konrad-Adenauer-Straße, Sankt Augustin, 53757, Germany.

*Corresponding author(s). E-mail(s): zeyd.boukhers@fit.fraunhofer.de;
Contributing authors: nagarajbahubali@uni-koblenz.de;

Abstract

In the academic world, the number of scientists grows every year and so does the number of authors sharing the same names. Consequently, it challenging to assign newly published papers to their respective authors. Therefore, Author Name Ambiguity (ANA) is considered a critical open problem in digital libraries. This paper proposes an Author Name Disambiguation (AND) approach that links author names to their real-world entities by leveraging their co-authors and domain of research. To this end, we use data collected from the DBLP repository that contains more than 5 million bibliographic records authored by around 2.6 million co-authors. Our approach first groups authors who share the same last names and same first name initials. The author within each group is identified by capturing the relation with his/her co-authors and area of research, represented by the titles of the validated publications of the corresponding author. To this end, we train a neural network model that learns from the representations of the co-authors and titles. We validated the effectiveness of our approach by conducting extensive experiments on a large dataset.

Keywords: author name disambiguation, entity linkage, bibliographic data, neural networks, classification, DBLP

1 Introduction

AND is an important task in digital libraries that aims to properly link each publication to its respective co-authors so that author-level metrics can be accurately calculated and authors' publications can be easily found. However, this task is extremely challenging due to the high number of authors sharing the same names. In this paper, *author name* denotes a sequence of characters referring to one or several authors [1], whereas *author* refers to a unique person authoring at least one publication and cannot be identified only by his/her *author name* [2] but rather with the support of other identifiers such as ORCID, ResearchGate ID and Semantic Scholar author ID.

Although relying on these identifiers almost eliminates any chance of mislinking a publication to its appropriate author, most bibliographic sources do not include such identifiers. This is because not all of the authors are keen to use these identifiers and if they are, there is no procedure or policy to include their identifiers when they are cited. Therefore, in bibliographic data (e.g. references), authors are commonly referred to by their names only. Considering the high number of authors sharing the same names (i.e. homonymy), it is difficult to link the names in bibliographic sources to their real-world authors especially when the source of the reference is not available or does not provide indicators of the author's identity. The problem is more critical when names are substituted by their initials to save space, and when they are erroneous due to wrong manual editing. Disciplines like social sciences and humanities suffer more from this problem as most of the publishers are small and mid-sized and cannot ensure the continuous integrity of the bibliographic data.

Table 1 demonstrates real examples of reference strings covering the above-mentioned problems. The homonomy issue shows an example of two different papers citing the name *J M Lee* which refers to two different authors. In this case, it is not possible to disambiguate the two authors without leveraging other features. The Synonymy issue shows an example of the same author *Jang Myung Lee* cited differently in two different papers as *Jang Myung Lee* and *J Lee*. Synonymy is a serious issue in author name disambiguation as it requires the awareness of all name variates of the given author. Moreover, some name variates might be shared by other authors, which increases homonymy.

Since these problems are known for decades, several studies [1–9] have been conducted using different machine learning approaches. This problem is

[1] It is estimated that about 114 million people share 300 common names.

[2] In the DBLP database, there are 27 exact matches of 'Chen Li', 23 reverse matches and more than 1000 partial matches

[3] Xu, Zhihao, et al. "Teleoperating a formation of car-like rovers under time delays." Proceedings of the 30th Chinese Control Conference. IEEE, 2011.

[4] Shi, Pu, Jianning Hua, and Yiwen Zhao. "Posture-based virtual force feedback control for teleoperated manipulator system." 2010 8th World Congress on Intelligent Control and Automation. IEEE, 2010.

[5] Xu, Zhihao, Lei Ma, and Klaus Schilling. "Passive bilateral teleoperation of a car-like mobile robot." 2009 17th Mediterranean Conference on Control and Automation. IEEE, 2009.

[6] Lu, Ching-Hsi, Hong-Yang Hsu, and Lei Wang. "A new contrast enhancement technique by adaptively increasing the value of histogram." 2009 IEEE international workshop on imaging systems and techniques. IEEE, 2009.

Table 1 Illustrative examples of author name ambiguity and incorrect author names

Issue Type	Source	Citations
Synonyms	See [3]	T. Jin, **J. Lee**, and H. Hashimoto, "Internet-based obstacle avoidance of mobile robot using a force-reflection," in Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, (Sendai, Japan), pp. 3418–3423, October 2004.
	See [4]	TasSeok Jin, **JangMyung Lee**, and Hideki Hashimoto, "Internet-based obstacle avoidance of mobile robot using a force-reflection," IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3418-3423. 2004.
Homonyms	See [5]	T.S. Jin, **J.M. Lee**, and H. Hashimoto. Internet-based obstacle avoidance of mobile robot using a force-reflection. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3418–3423, Sendai, Japan, October 2004.
	See [6]	H-J Kim, **J-M Lee**, J-A Lee, S-G Oh, W-Y Kim, "Contrast Enhancement Using Adaptively Modified Histogram Equalization", Lecture Notes in Computer Science, Vol.4319, pp.1150 - 1158, Dec. 2006.

often tackled using supervised approaches such as Support Vector Machine (SVM) [10], Bayesian Classification [7] and Neural networks (NN) [11]. These approaches rely on the matching between publications and authors which are verified either manually or automatically. Unsupervised approaches [12–14] have also been used to assess the similarity between a pair of papers. Other unsupervised approaches are also used to estimate the number of co-authors sharing the same name [15] and decide whether new records can be assigned to an existing author or a new one [6]. Due to the continuous increase of publications, each of which cites tens of other publications and the difficulty to label this streaming data, semi-supervised approaches [16, 17] were also employed. Recent approaches [18, 19] leveraged the outstanding efficiency of deep learning on different domains to exploit the relationship among publications using network embedding. All these approaches use the available publication data about authors such as titles, venues, year of publication and affiliation. Some of these approaches are currently integrated into different bibliographic systems. However, all of them require an exhausting manual correction to reach an acceptable accuracy. In addition, most of these approaches rely on the metadata extracted from the papers which are supposed to be correct and complete. In real scenarios, the source of the paper is not always easy to find and only the reference is available.

In this paper, which builds upon our earlier work [20], we aim to employ bibliographic data consisting of publication records to link each author's name in unseen records to their appropriate real-world authors (i.e. DBLP identifiers) by leveraging their co-authors and area of research embedded in the publication title and source. Note that the goal of this paper is to disambiguate author names in newly published papers that are not recorded in any bibliographic database. Therefore, all records that are considered unseen are discarded from the bibliographic data and used only for testing the approach. The assumption is that any author is most likely to publish articles in specific fields of research. Therefore, we employ articles' titles and sources (i.e. Journal, Booktitle, etc.) to bring authors close to their fields of research represented by the titles and sources of publications. We also assume that authors who already published together are more likely to continue collaborating and publishing other papers.

For the goal mentioned above, our proposed model is trained on a bibliographic collection obtained from DBLP, where a sample consists of a target author, pair of co-authors, title and source. For co-authors, the input is a vector representation obtained by applying Char2Vec which returns character-level embedding of words. For title and source, the BERT model is used to capture the semantic representations of the sequence of words. Our model is trained and tested on a challenging dataset, where thousands of authors share the same atomic name variate. The main contributions of this paper are:

- We proposed a novel approach for author name disambiguation using semantic and symbolic representations of titles, sources, and co-authors.
- We provided a statistical overview of the problem of author name ambiguity.
- We conducted experiments on challenging datasets simulating a critical scenario.
- The obtained results and the comparison against baseline approaches demonstrate the effectiveness of our model in disambiguating author names.

The rest of the paper is organized as follows. Section 2 briefly presents related work. Section 3 describes the proposed framework. Section 4 presents the dataset, implementation details and the obtained results of the proposed model. Finally, Section 5 concludes the paper and gives insights into future work.

2 Related Work

In this section, we discuss recent approaches softly categorized into three categories, namely unsupervised-, supervised- and graph-based;

2.1 Unsupervised-based:

Most of the studies treat the problem of author name ambiguity as an unsupervised task [6, 9, 9, 13, 15] using algorithms like DBSCAN [9] and agglomerative clustering [21]. Liu et al. [12] and Kim et al. [13] rely on the similarity between a pair of records with the same name to disambiguate author names on the PubMed dataset. Zhang et al. [15] used Recurrent Neural Network (RNN) to estimate the number of unique authors in the Aminer dataset. This process is followed by manual annotation. In this direction, Ferreira et al. [22] have proposed a two-phase approach applied to the DBLP dataset, where the first one is obtaining clusters of authorship records and then disambiguation is applied to each cluster. Wu et al. [21] fused features such as affiliation and content of papers using Shannon's entropy to obtain a matrix representing pairwise correlations of papers which is in return used by Hierarchical Agglomerative Clustering (HAC) to disambiguate author names on Arnetminer dataset. Similar features have been employed by other approaches [23, 24].

2.2 Supervised-based:

Supervised approaches [7, 10, 11, 25, 26] are also widely used but mainly only after applying to block that gathers authors sharing the same names together. Han et al. [10] present two supervised learning approaches to disambiguate authors in cited references. Given a reference, the first approach uses the Naive Bayes model to find the author class with the maximal posterior probability of being the author of the cited reference. The second approach uses SVM to classify references from DBLP to their appropriate authors. Sun et al. [26] employ heuristic features like the percentage of citations gathered by the top name variations for an author to disambiguate common author names. Neural networks are also used [11] to verify if two references are close enough to be authored by the same target author or not. Hourrane et al. [27] propose a corpus-based approach that uses word embeddings to compute the similarity between cited references. In [28], an Entity Resolution system called the DEEPER is proposed. It uses a combination of bi-directional recurrent neural networks (BRNN) along with Long Short Term Memory (LSTM) as the hidden units to generate a distributed representation for each tuple to capture the similarities between them. Zhang et al. [7] proposed an online Bayesian approach to identify authors with ambiguous names and as a case study, bibliographic data in a temporal stream format is used and the disambiguation is resolved by partitioning the papers into homogeneous groups.

2.3 Graph-based:

As bibliographic data can be viewed as a graph of citations, several approaches have leveraged this property to overcome the problem of author name ambiguation [18, 19, 29, 30]. Hoffart et al. [29] present a method for collective disambiguation of author names, which harnesses the context from a knowledge base and uses a new form of coherence graph. Their method generates a weighted graph of the candidate entities and mentions to compute a dense sub-graph that approximates the best entity-mention mapping. Xianpei et al. [30] aim to improve the traditional entity linking method by proposing a graph-based collective entity linking approach that can model and exploit the global interdependence, i.e., the mutual dependence between the entities. In [18], the problem of author name ambiguity is overcome using relational information considering three graphs: person-person, person-document and document-document. The task becomes then a graph clustering task with the goal that each cluster contains documents authored by a unique real-world author. For each ambiguous name, Xu et al. [19] build a network of papers with multiple relationships. A network-embedding method is proposed to learn paper representations, where the gap between positive and negative edges is optimized. Further, HDBSCAN is used to cluster paper representations into disjoint sets such that each set contains all papers of a unique real-world author.

3 Approach:

In this paper, AND is designed using a bibliographic dataset $\mathcal{D} = \{d_i\}_{i=1}^{N}$, consisting of N bibliographic records, where each record d_i refers to a unique publication such that $d_i = \{t_i, s_i, \langle a_{i,u}, \delta_{i,u}\rangle_{u=1}^{\omega_i}\}$. Here, t_i and s_i denote the *title* and *source* of the record, respectively. $a_{i,u}$ and $\delta_{i,u}$ refer to the *uth* author and its corresponding name, respectively, among ω_i co-authors of d_i. Let $\Delta = \{\delta(m)\}_{m=1}^{M}$ be a set of M unique author names in D shared by a set of L unique authors $\mathcal{A} = \{a(l)\}_{l=1}^{L}$ co-authoring all records in D, where $L >> M$. Note that each author name $\delta(m)$ might refer to one or more authors in $\mathcal{A}$ and each author $a(l)$ might be referred to by one or two author names in Δ. This is because we consider two variates for each author as it might occur differently in different papers. For example, the author "*Rachid Deriche*" is assigned to two elements in Δ, namely "*Rachid Deriche*" and "*R. Deriche*".

Given a reference record $d^* \notin \mathcal{D}$, the goal of our approach is to link each author name $\delta_u^* \in \Delta$ that occurs in d^* to the appropriate author in $\mathcal{A}$ by leveraging t^*, s^* and $\{\delta_u^*\}_{u=1}^{\omega^*}$. Figure 1 illustrates an overview of our proposed approach. First, the approach computes the correspondence frequency $\delta_u^*\mathbf{R}\mathcal{A}$ that returns the number of authors in $\mathcal{A}$ corresponding to δ_u^*. $\delta_u^*\mathbf{R}\mathcal{A} = 0$ indicates that δ_u^* corresponds to a new author $a(\text{new}) \notin \mathcal{A}$. $\delta_u^*\mathbf{R}\mathcal{A} = 1$ indicates that δ_u^* corresponds to only one author $a(l) \in \mathcal{A}$. In this case, we directly assign δ_u^* to $a(l)$ and no further processing is necessary. Note that in this case, δ_u^* might also refer to a new author $a(\text{new}) \notin \mathcal{A}$ who has the same name as an existing author $a(l) \in \mathcal{A}$. However, our approach does not handle this situation. Please refer to Section 4.3 that lists the limitation of the proposed approach.

The goal of this paper is to handle the case of $\delta_u^*\mathbf{R}\mathcal{A} > 1$ which indicates that δ_u^* can refer to more than one author. To this end, the approach extracts the atomic name variate from the author name δ_u^*. For example, for the author name $\delta_u^* =$ "*Lei Wang*", the atomic name variate is $\overline{\delta_u^*} =$ "*L Wang*". Let $\overline{\delta_u^*}$ correspond to $\overline{\delta_\mu}$ which denotes the *μth* atomic name variate among K possible name variates. Afterwards, the corresponding Neural Network model $\theta_\mu \in \Theta = \{\theta_k\}_{k=1}^{K}$ is picked to distinguish between all authors $\mathcal{A}_\mu = \{a(l_\mu)\}_{l_\mu=1}^{L_\mu}$ who share the same name variate $\overline{\delta_\mu}$.

3.1 Model Architecture

The Neural Network (NN) model θ_μ takes as input the attributes of d^*, namely the first name of the target author $\delta_u^{*\text{first-name}}$, full names of two co-authors δ_p^* and δ_j^*, title t^* and source s^*. Figure 2 illustrates the architecture of θ_μ, with an output layer of length L_k corresponding to the number of unique authors in $\mathcal{A}_\mu$ who have the same atomic name variate δ_k. As shown in Figure 2, θ_μ takes two inputs $\mathbf{x}_{\mu,\mathbf{1}}$ and $\mathbf{x}_{\mu,\mathbf{2}}$, such that:

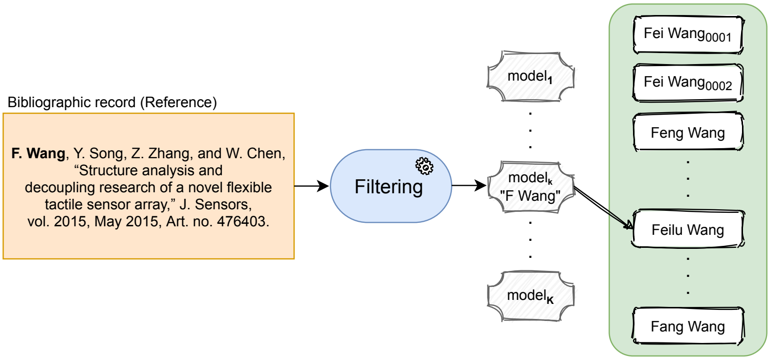
Bibliographic record (Reference)
F. Wang, Y. Song, Z. Zhang, and W. Chen, "Structure analysis and decoupling research of a novel flexible tactile sensor array," J. Sensors, vol. 2015, May 2015, Art. no. 476403.
Filtering
model$_1$
model$_k$ "F Wang"
model$_K$
Fei Wang$_{0001}$
Fei Wang$_{0002}$
Feng Wang
Feilu Wang
Fang Wang

Fig. 1 An illustration of the task for linking a name mentioned in the reference string with the corresponding DBLP author entity.

$$\begin{aligned} \mathbf{x}_{\mu,\mathbf{1}} &= \text{char2vec}(\delta_u^{*\text{first-name}}) \bigoplus \frac{1}{2}\left(\text{char2vec}(\delta_p^*) + \text{char2vec}(\delta_j^*)\right), \\ \mathbf{x}_{\mu,\mathbf{2}} &= \frac{1}{2}\left(\text{bert}(t^*) + \text{bert}(s^*)\right), \end{aligned} \tag{1}$$

where char2vec($\mathbf{w}$) returns a vector representation of length 200 generated using *Char2Vec* [31], which provides a symbolic representation of w. bert($\mathbf{w}$) returns a vector representation of each token in $\mathbf{w}$ w.r.t its context in the sentence. This representation of length 786 is generated using BERT [32]. The goal of separating the two inputs is to overcome the sparseness of content embedding and force the model to emphasise more on target author representation.

All the hidden layers possess a ReLU activation function, whereas the output is a Softmax classifier. Since the model has to classify thousands of classes, each of which is represented with very few samples, 50% of the units in the last hidden layers are dropped out during training to avoid over-fitting. Furthermore, the number of publications significantly differs from one author to another. Therefore, each class (i.e. the author) is weighted according to its number of samples (i.e. publications). The model is trained with *adam* optimizer and sparse categorical cross-entropy loss function. Our empirical analysis showed that the best performance was achieved with this architecture and these parameters, which were obtained through grid search.

3.2 Author name representation

The names of authors do not hold any specific semantic nature as they are simply a specific sequence of characters referring to one or more persons. Therefore, we need a model that can encode words based on the order and distribution of characters such that author names with a similar name spellings are encoded closely, assuming possible manual editing errors of cited papers.

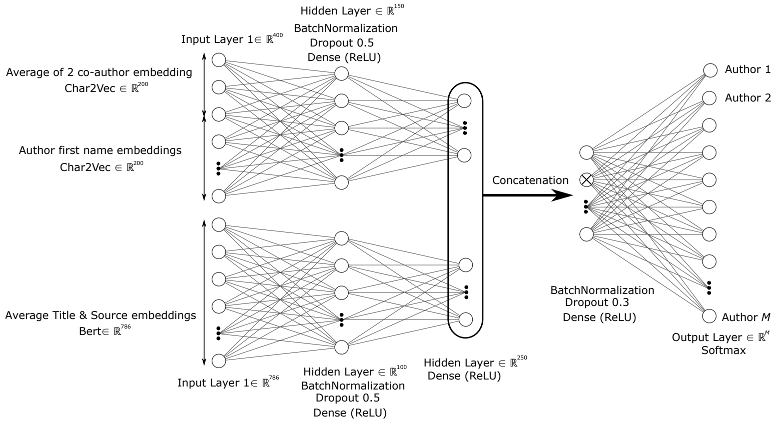
Hidden Layer ∈ $\mathbb{R}^{150}$
BatchNormalization
Dropout 0.5
Dense (ReLU)
Input Layer 1∈ $\mathbb{R}^{400}$
Average of 2 co-author embedding
Char2Vec ∈ $\mathbb{R}^{200}$
Author first name embeddings
Char2Vec ∈ $\mathbb{R}^{200}$
Average Title & Source embeddings
Bert∈ $\mathbb{R}^{786}$
Input Layer 1∈ $\mathbb{R}^{786}$
Hidden Layer ∈ $\mathbb{R}^{100}$
BatchNormalization
Dropout 0.5
Dense (ReLU)
Hidden Layer ∈ $\mathbb{R}^{250}$
Dense (ReLU)
Concatenation
BatchNormalization
Dropout 0.3
Dense (ReLU)
Author 1
Author 2
Author *M*
Output Layer ∈ $\mathbb{R}^{M}$
Softmax

Fig. 2 The architecture of our model.

Chars2vec is a powerful NN-based language model that is preferred when the text consists of abbreviations, typos, etc. It captures the non - vocabulary words and places words with similar spelling closer in the vector space. This model uses a fixed list of characters for word vectorization, where a one-hot encoding represents each character.

3.3 Source and Title embedding

The source (e.g. journal names and book titles) of reference can provide a hint about the area of research of the given reference. In addition, the title is a meaningful sentence that embeds the specific topic of the reference. Therefore, we used these two features to capture the research area of the author. Contrary to the author's name, the goal here is to capture the context of the sequences of words forming the title and source. Therefore, we employed the pre-trained BERT model [32] to obtain sentence embeddings of both the title and source.

3.4 Model Training

Given the training set $\mathcal{D}_\mu \subset \mathcal{D}$ that corresponds to the subset of bibliographic records authored by authors having the atomic name variate $\overline{\delta_\mu}$, $d_{i_\mu} \in \mathcal{D}_\mu$ generates ω_{i_μ} training samples $\langle \delta_\mu, \delta_{i_\mu,p}, \delta_{i_\mu,j}, t_{i_\mu}, s_{i_\mu} \rangle_{p=1}^{\omega_{i_\mu}}$, where $\delta_{i_\mu,j}$ is a random co-author of d_{i_μ} and might be also the same author name as $\delta_{i_\mu,p}$ and/or δ_μ. Note also that we consider one combination where $\delta_{i_\mu,p} = \delta_\mu$. In order to train the model with the other common name variate where the first name is substituted with its initial, for each sample, we generate another version with name variates $\langle \overline{\delta_\mu}, \overline{\delta_{i_\mu,p}}, \overline{\delta_{i_\mu,j}}, t_{i_\mu}, s_{i_\mu} \rangle$. Consequently, each bibliographic record is fed into the model $2 \times \omega_{i_\mu}$ times.

Since the third co-author $\delta_{i_\mu,p}$ is randomly assigned to the training sample among ω_{i_μ} co-authors d_{i_μ}, we randomly reassign it after Y epochs. In addition to lower training complexity, this has shown in the conducted experiments a slightly better result than training the model at each epoch with samples of all possible co-author pairs p and j.

3.5 Model Tuning

For each training epoch, *WhoIs* model fine-tunes the parameters to predict the appropriate target author. The performance of the model is considerably influenced by the number of epochs set to train. Specifically, a low epoch count may lead to underfitting. Whereas, a high epoch count may lead to over-fitting. To avoid this, we enabled early stopping, which allows the model to specify an arbitrarily large number of epochs.

Keras supports early stopping of the training via a callback called *EarlyStopping*. This callback is configured with the help of the *monitor* argument which allows setting the validation loss. With this setup, the model receives a trigger to halt the training when it observes no more improvement in the validation loss.

Often, the very first indication of no more improvement in the validation loss would not be the right epoch to stop training; because the model may start improving again after passing through a few more epochs. We overcome this by adding a delay to the trigger in terms of consecutive epochs count on which, we can wait to observe no more improvement. A delay is added by setting the *patience* argument to an appropriate value. *patience* in *WhoIs* is set to 50, so that the model only halts when the validation loss stops getting better for the past 50 consecutive epochs.

3.6 Model checkpoint

Although *WhoIs* stops the training process when it achieves a minimum validation loss, the model obtained at the end of the training may not give the best accuracy on validation data. To account for this, Keras provides an additional callback called *ModelCheckpoint*. This callback is configured with the help of another *monitor* argument. We have set the *monitor* to monitor the validation accuracy. With this setup, the model updates the weights only when it observes better validation accuracy compared to earlier epochs. Eventually, we end up persisting in the best state of the model with respect to the best validation accuracy.

3.7 Prediction:

Given the new bibliographic record $d^* = \{t^*, s^*, \langle \delta^*_u \rangle_{u=1}^{\omega^*}\}$, the goal is to disambiguate the author name δ^*_{target} which is shared by more than one author $(\delta^*_{\text{target}} \mathbf{R} \mathcal{A} > 1)$. To this end, Y samples $S_{y=1}^{Y}$ are generated for all possible pairs of co-author names p and j: $\langle \delta^*_{\text{target}}, \delta^*_p, \delta^*_j, t^*, s^* \rangle_{p=1,j=1}^{\omega^*,\omega^*}$, where $Y = \mathrm{C}(\omega^* + 1, 2)$, i.e. the combination of $\omega^* + 1$ authors taken 2 at a time, and δ^*_u can be a full or abbreviated author name. All the Y samples are fed to the corresponding model θ_μ, where the target author a_{target} of the target name δ^*_{target} is predicted as follows:

$$a_{target} = \underset{1 \cdots L_\mu}{argmax} \left(\theta_\mu(S_1) \oplus \theta_\mu(S_2) \oplus \cdots \oplus \theta_\mu(S_Y)\right), \tag{2}$$

where $\theta_\mu(S_y)$ returns a probability vector of length L_μ with each element l_μ denotes the probability of the author name δ^*_{target} to be the author a_{l_μ}.

4 Experiments

This section presents the experimental results of the proposed approach to the DBLP dataset.

4.1 Dataset

The following datasets are widely used to evaluate author name disambiguation approaches but the results on these datasets cannot reflect the results on real scenario streaming data.

- **ORCID [7]:** it is the largest accurate dataset as the publication is assigned to the author only after authorship claim or another rigorous authorship confirmation. However, this accuracy comes at the cost of the number of assignments. Our investigation shows that most of the registered authors are not assigned to any publication and an important number of authors are not even registered. This is because most of the authors are not keen to claim their publications due to several reasons.
- **KDD Cup 2013 [8]:** it is a large dataset that consists of 2.5M papers authored by 250K authors. All author metadata are available including affiliation.
- **Manually labelled (e.g. PENN [9], QIAN [10], AMINER [11], KISTI [12]):** These datasets are supposed to be very accurate since they are manually labelled. However, this process is expensive and time-consuming and, therefore, it can cover only a small portion of authors who share the same names.

In this work, we collected our dataset from the DBLP bibliographic repository[13]. The DBLP version of July 2020 contains 5.4 million bibliographic records such as conference papers, articles, thesis, etc., from various fields of research. As stated by the maintainers of DBLP [14], the accuracy of the data is not guaranteed. However, a lot of effort is put into manually disambiguating homonym cases when reported by other users. Consequently, we are aware of possible homonym cases that are not resolved yet. From the repository, we collected only records of publications published in journals and proceedings. Each record in this collection represents metadata information of a publication with one or more authors, title, journal, year of publication and a few other attributes. The availability of these attributes differs from one reference

[7] https://figshare.com/articles/ORCID_Public_Data_File_2017/5479792

[8] https://www.kaggle.com/c/kdd-cup-2013-author-paper-identification-challenge

[9] http://clgiles.ist.psu.edu/data/nameset_author-disamb.tar.zip

[10] https://github.com/yaya213/DBLP-Name-Disambiguation-Dataset

[11] http://arnetminer.org/lab-datasets/disambiguation/rich-author-disambiguation-data.zip

[12] http://www.lbd.dcc.ufmg.br/lbd/collections/disambiguation/DBLP.tar.gz/at_download/file

[13] https://dblp.uni-trier.de/xml/ (July 2020)

[14] https://dblp.org/faq/How+accurate+is+the+data+in+dblp.html

Table 2 Statistical details of the used DBLP collection.

# of records	5258623
# of unique authors	2665634
# of unique author names	2613577
# of unique atomic name variates	1555517

to another. Also, the authors in DBLP who share the same name have a suffix number to differentiate them. For instance, the authors with the same name 'Bing Li' are given suffixes such as 'Bing Li 0001', and 'Bing Li 0002'. The statistical details of the used DBLP collection are shown in Table 2.

Figure 3 indicates that the majority of target authors in the sub-collections (each sub-collection includes all records of authors with the same name) have distinct full names. However, a considerable number of them share full names, leading to a significant challenge, particularly when multiple authors (e.g. over 80 in 4 out of 5 sub-collections) share the same full name but have an unequal number of publications. In such cases, it becomes more challenging to differentiate these authors from the dominant author with the same name.

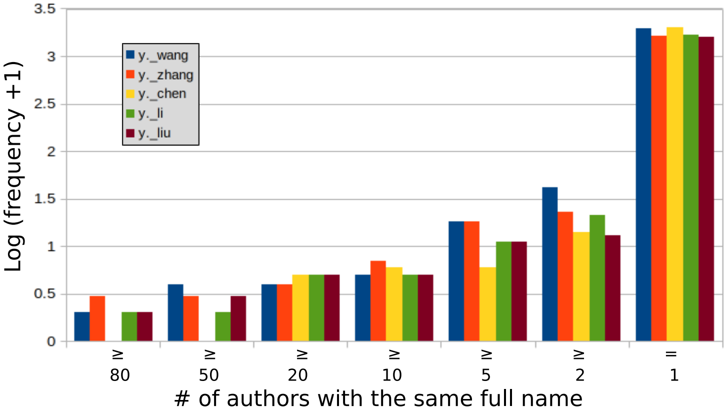

Fig. 3 The *log* frequency of authors sharing the same full name for the top five sub-collections.

Figure 4 illustrates the log frequency of bibliographic records with the same full name in the top five sub-collections used in this paper. As illustrated, in all sub-collections, the target authors of around half of the records authored a few records (less than 5) and have unique names. Although it is simple to distinguish these authors when their full names occur, it is extremely challenging to recognize them among more than 2000 authors sharing the same atomic name variate due to the unbalance of records with the other authors.

Figure 5 shows the frequency of authors sharing the same names and the same atomic name variates. As can be seen, the problem is more critical when the authors are cited with their atomic name variate as there are five atomic name variates shared by around 11.5k authors. This makes the problem of disambiguation critical because not only targets authors who might share the

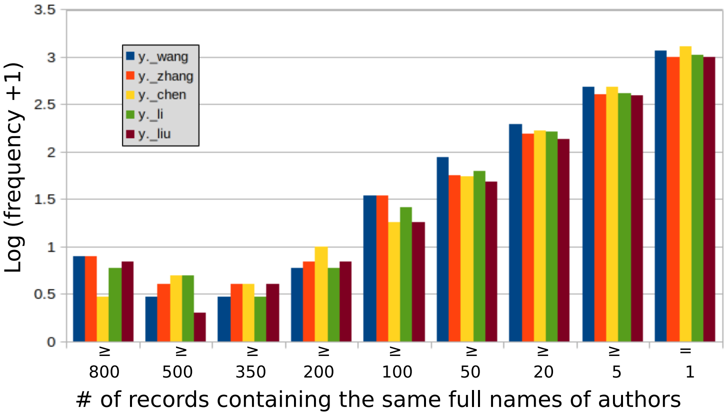

Fig. 4 The *log* frequency of records with the same full name of the target author for the top five sub-collections.

same atomic name variate but also their co-authors. For instance, we observed publications authored by the pair of co-authors having the atomic name variates: *Y. Wang* and *Y. Zhang*. However, they refer to different *Y. Wang* and *Y. Zhang* pairs of real-world authors.

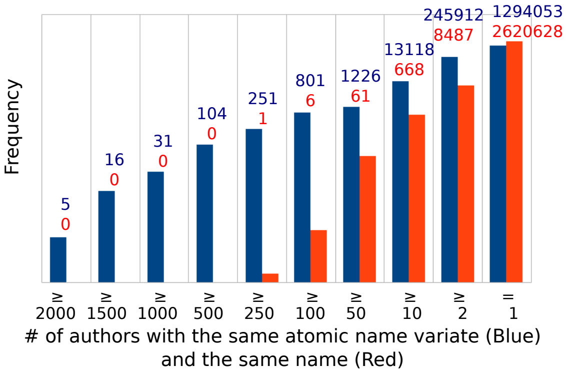

Fig. 5 Frequency of authors sharing the same atomic name variate (Blue) / the same full name (Red).

Since our approach gathers authors with the same name variates, 261464 models are required to disambiguate all author names in our collection. Therefore, we present in this paper the experimental results on 5 models corresponding to the highest number of authors sharing the same name variates. Table 3 presents statistical details of the five sub-collections which demonstrates the challenges inherent in author name disambiguation in real-world scenarios. # **R2A** for instance, in some publications, two co-authors have the same exact names. This makes the disambiguation more difficult as these authors share not only their names but also co-authors and papers.

To ensure a credible evaluation and result reproducibility in real scenarios, we split the records in each sub-collection into a training set ($\sim$ 70%),

Table 3 Statistical details of the top 5 sub-collections of authors sharing the same atomic name variates, where # **ANV** is the corresponding atomic name variate, # **UTA** is the number of unique target authors, # **RCD** is the number of bibliographic records, # **UCA** is the number of unique co-author full names, # **UAN** is the number of unique target author full names, # **R2A** is the number of records with two co-authors of the same record having the same names or the same atomic name variates and # **R3A** is the number of records with three co-authors of the same record having the same names or the same atomic name variates. For # **R2A** and # **R3A**, it is not necessary that the authors have the same name / atomic name variate as the target author but most probably.

	'Y Wang'	'Y Zhang'	'Y Chen'	'Y Li'	'Y Liu'
# **UTA**	2601	2285	2260	2166	2142
# **RCD**	37409	33639	26155	29154	27691
# **UCA**	43199	39389	33461	35765	33754
# **UAN**	2005	1667	2034	1734	1606
# **R2A**	582	598	316	372	338
# **R3A**	13	12	4	4	3

validation set ($\sim$ 15%) and testing set ($\sim$ 15%) in terms of records/target author. Specifically, for each target author, we randomly split the corresponding records. If the target author did not author enough publications for the split, we prioritize the training set, then validation and finally the test set. Consequently, the number of samples is not necessarily split according to $70 : 15 : 15$ as the number of co-authors differs among publications. Moreover, it is highly likely that the records of a unique target author are completely different among the three sets. Consequently, it is difficult for the model to recognize the appropriate author only from his/her co-authors and research area. However, we believe that this is more realistic and a perfect simulation of the real scenario.

To account for possible name variates, each input sample of full names is duplicated, where the duplicate down sample full names of all co-authors to atomic name variates. Note that this is applied to training, validation and test sets. The goal is to let the model capture all name variates for each author and his/her co-authors. In none of the sets, the variates are mixed in a single sample as we assume that this case is very less likely to occur in the real world. The experiments were conducted on a machine with the following specifications:

- Processor: AMD Ryzen Threadripper 1950X 16-Core
- RAM: 12 GB
- Graphics card: NVIDIA Titan V GV100

The algorithm was implemented in Python 3.7 using the TensorFlow library.

4.2 Results

The existing AND approaches use different datasets to design and evaluate their models. This lead to different assumptions and challenge disparity. Unfortunately, the codes to reproduce the results of these approaches are not available or easily accessed [4]. Therefore, it is not possible to fairly compare *WhoIs* against baseline approaches. For future work, our code and the used datasets are publicly available [15].

[15] https://whois.ai-research.net

Table 4 Detailed results of *WhoIs* on the sub-collections corresponding to the top five of authors sharing the same atomic name variates in the DBLP repository. The results are presented in terms of Micro average precision (**MiAP**), Macro average precision (**MaAP**), Micro average recall (**MiAR**), Macro average recall (**MaAR**), Micro average F1-score (**MiAF1**) and Macro average F1-score (**MaAF1**). **ANV** denotes that only atomic name variates were used for all target authors and all their co-authors.

	'Y Wang'	'Y Zhang'	'Y Chen'	'Y Li'	'Y Liu'
MaAP(ANV)	0.226	0.212	0.255	0.193	0.218
MaAP(All)	0.387	0.351	0.404	0.342	0.347
MaAR(ANV)	0.299	0.276	0.301	0.229	0.267
MaAR(All)	0.433	0.383	0.409	0.339	0.361
MaAF1(ANV)	0.239	0.220	0.258	0.195	0.223
MaAF1(All)	0.385	0.342	0.383	0.321	0.332
MiAF1(ANV)	0.274	0.278	0.366	0.260	0.322
MiAF1(All)	0.501	0.482	0.561	0.492	0.504

Table 4 presents the result of *WhoIs* on the sub-collections presented in Table 3. The label *All* in the table denotes that all samples were predicted twice, one with full names of the target author and its co-authors and another time with only their atomic name variates, whereas the label *ANV* denotes that only samples with atomic names are predicted. The obtained results show that an important number of publications are not properly assigned to their appropriate authors. This is due to the properties of the sub-collections which were discussed above and statistically presented in Table 3. For example, 1) two authors with the same common name authoring a single publication. 2) more than one author with the same common atomic name variate authoring a single publication, 3) number of authors with the same full name, 4) the uncertainty of the accuracy of the dataset, etc.

Although the comparison is difficult and cannot be completely fair, we compare *WhoIs* to other state-of-the-art approaches, whose results are reported in [18]. These results are obtained on a collection from CiteSeerX [16] that contains records of authors with the name / atomic name variate '*Y Chen*'. This collection consists of 848 complete documents authored by 71 distinct authors. We picked this name for comparison because of two reasons; 1) the number of authors sharing this name is among the top five as shown in Table 3 and 2) All methods cited in [18] could not achieve a good result. We applied *WhoIs* on this collection by randomly splitting the records into 70% for training, 15% for validation and 15% for testing. The results are shown in Table 5. Note that in our collection, we consider way more records and distinct authors (see Table 3) and we use only reference attributes (i.e. co-authors, title and source).

As the results presented in Table 5 show, *WhoIs* outperforms other methods in resolving the disambiguation of the author name '*Y Chen*' on the CiteSeerX dataset, which is a relatively small dataset and does not really reflect the performance of all presented approaches in real scenarios. The disparity between the results shown in Table 4 and Table 5 demonstrates that the existing benchmark datasets are manually prepared for the sake of accuracy. However, this leads to covering a very small portion of records whose authors share similar

[16] http://clgiles.ist.psu.edu/data/

Table 5 Comparison between *WhoIs* and other baseline methods on CiteSeerX dataset in terms of Macro F1 score as reported in [18]. **ANV** denotes that only atomic name variates were used for all target authors and all their co-authors.

	Macro ALL/ANV	Micro ALL/ANV
WhoIs	**0.713** / **0.702**	0.873 / 0.861
NDAG [18]	0.367	N/A
GF [33]	0.439	N/A
DeepWalk [34]	0.118	N/A
LINE [35]	0.193	N/A
Node2Vec [36]	0.058	N/A
PTE [37]	0.199	N/A
GL4 [38]	0.385	N/A
Rand [18]	0.069	N/A
AuthorList [18]	0.325	N/A
AuthorList-NNMF [18]	0.355	N/A

names. This disparity confirms that author name disambiguation is still an open problem in digital libraries and far from being solved.

The obtained results of *WhoIs* illustrate the importance of relying on the research area of target authors and their co-authors to disambiguate their names. However, they trigger the need to encourage all authors to use different author identifiers such as ORCID [39] in their publications as the automatic approaches are not able to provide a perfect result mainly due to the complexity of the problem.

4.3 Limitations and obstacles of *WhoIs*:

WhoIs demonstrated a satisfactory result and outperformed state-of-the-art approaches on a challenging dataset. However, the approach faces several obstacles that will be addressed in our future works. In the following, we list the limitations of the proposed approach:

- New authors cannot be properly handled by our approach, where a confidence threshold is set to decide whether the input corresponds to a new author or an existing one. To our knowledge, none of the existing supervised approaches is capable to handle this situation.
- Commonly, authors found new collaborations which lead to new co-authorship. Our approach cannot benefit from the occurrence of new co-combinations of co-authors as they were never seen during training.
  Planned solution: We will train an independent model to embed the author's discipline using his/her known publications. With this, we assume that authors working in the same area of research will be put close to each other even if they did not publish a paper together, the model would be able to capture the potential co-authorship between a pair of authors in terms of their area of research.
- Authors continuously extend their research expertise by co-authoring new publications in relatively different disciplines. This means that the titles and journals are not discriminative anymore. Consequently, it is hard for our approach to disambiguate authors holding common names.

Planned solution: we plan to determine the author's areas of research by mining domain-specific keywords from the entire paper instead of its title assuming that the author uses similar keywords/writing styles even in different research areas with gradual changes which can be captured by the model.

- There are a lot of models that have to be trained to disambiguate all authors in the DBLP repository.
- Commonly, the number of samples is very small compared to the number of classes (i.e. authors sharing the same atomic name variate) which leads to overfitting the model.

  Planned solution: we plan to follow a reverse strategy of disambiguation. Instead of employing the co-authors of the target author, we will employ their co-authors aiming to find the target author among them. We aim also to learn co-author representation by employing their co-authors to help resolve the disambiguation of the target author's name.
- As mentioned earlier and stated by the maintainers of the platform [17], the accuracy of the DBLP repository is not guaranteed.

5 Conclusion

We presented in this paper a comprehensive overview of the problem of AND. To overcome this problem, we proposed a novel framework that consists of a lot of supervised models. Each of these models is dedicated to distinguishing among authors who share the same atomic name variate (i.e. first name initial and last name) by leveraging the co-authors and the titles and sources of their known publications. The experiments on challenging and real-scenario datasets have shown promising and satisfactory results on AND. We also demonstrated the limitations and challenges that are inherent in this process.

To overcome some of these limitations and challenges, we plan for future work to exploit citation graphs so that author names can be linked to real-world entities by employing the co-authors of their co-authors. We assume that using this reverse process, the identity of the target author can be found among the co-authors of his/her co-authors. We plan also to learn the research area of co-authors in order to overcome the issue of new co-authorships.

References

[1] Müller, M.-C.: Semantic author name disambiguation with word embeddings. In: International Conference on Theory and Practice of Digital Libraries, pp. 300–311 (2017). Springer

[2] Kim, K., Sefid, A., Weinberg, B.A., Giles, C.L.: A web service for author name disambiguation in scholarly databases. In: 2018 IEEE International Conference on Web Services (ICWS), pp. 265–273 (2018). IEEE

[17] https://dblp.org/faq/How+accurate+is+the+data+in+dblp.html

[3] Foxcroft, J., d'Alessandro, A., Antonie, L.: Name2vec: Personal names embeddings. In: Canadian Conference on Artificial Intelligence, pp. 505–510 (2019). Springer

[4] Hussain, I., Asghar, S.: A survey of author name disambiguation techniques: 2010-2016. Knowledge Eng. Review **32**, 22 (2017)

[5] Ferreira, A.A., Gonçalves, M.A., Laender, A.H.: A brief survey of automatic methods for author name disambiguation. Acm Sigmod Record **41**(2), 15–26 (2012)

[6] Qian, Y., Zheng, Q., Sakai, T., Ye, J., Liu, J.: Dynamic author name disambiguation for growing digital libraries. Information Retrieval Journal **18**(5), 379–412 (2015)

[7] Zhang, B., Dundar, M., Al Hasan, M.: Bayesian non-exhaustive classification a case study: Online name disambiguation using temporal record streams. In: Proceedings of the 25th Acm International on Conference on Information and Knowledge Management, pp. 1341–1350 (2016)

[8] Khabsa, M., Treeratpituk, P., Giles, C.L.: Large scale author name disambiguation in digital libraries. In: 2014 IEEE International Conference on Big Data (Big Data), pp. 41–42 (2014). IEEE

[9] Khabsa, M., Treeratpituk, P., Giles, C.L.: Online person name disambiguation with constraints. In: Proceedings of the 15th Acm/ieee-cs Joint Conference on Digital Libraries, pp. 37–46 (2015)

[10] Han, H., Giles, L., Zha, H., Li, C., Tsioutsiouliklis, K.: Two supervised learning approaches for name disambiguation in author citations. In: Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries, 2004., pp. 296–305 (2004). IEEE

[11] Tran, H.N., Huynh, T., Do, T.: Author name disambiguation by using deep neural network. In: Asian Conference on Intelligent Information and Database Systems, pp. 123–132 (2014). Springer

[12] Liu, W., Islamaj Doğan, R., Kim, S., Comeau, D.C., Kim, W., Yeganova, L., Lu, Z., Wilbur, W.J.: Author name disambiguation for p ub m ed. Journal of the Association for Information Science and Technology **65**(4), 765–781 (2014)

[13] Kim, K., Sefid, A., Giles, C.L.: Learning cnf blocking for large-scale author name disambiguation. In: Proceedings of the First Workshop on Scholarly Document Processing, pp. 72–80 (2020)

[14] Fan, X., Wang, J., Pu, X., Zhou, L., Lv, B.: On graph-based name disambiguation. Journal of Data and Information Quality (JDIQ) **2**(2), 1–23 (2011)

[15] Zhang, Y., Zhang, F., Yao, P., Tang, J.: Name disambiguation in aminer: Clustering, maintenance, and human in the loop. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1002–1011 (2018)

[16] Louppe, G., Al-Natsheh, H.T., Susik, M., Maguire, E.J.: Ethnicity sensitive author disambiguation using semi-supervised learning. In: International Conference on Knowledge Engineering and the Semantic Web, pp. 272–287 (2016). Springer

[17] Zhao, J., Wang, P., Huang, K.: A semi-supervised approach for author disambiguation in kdd cup 2013. In: Proceedings of the 2013 KDD CUP 2013 Workshop, pp. 1–8 (2013)

[18] Zhang, B., Al Hasan, M.: Name disambiguation in anonymized graphs using network embedding. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 1239–1248 (2017)

[19] Xu, J., Shen, S., Li, D., Fu, Y.: A network-embedding based method for author disambiguation. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 1735–1738 (2018)

[20] Boukhers, Z., Asundi, N.B.: Whois? deep author name disambiguation using bibliographic data. In: Linking Theory and Practice of Digital Libraries: 26th International Conference on Theory and Practice of Digital Libraries, TPDL 2022, Padua, Italy, September 20–23, 2022, Proceedings, pp. 201–215 (2022). Springer

[21] Wu, H., Li, B., Pei, Y., He, J.: Unsupervised author disambiguation using dempster–shafer theory. Scientometrics **101**(3), 1955–1972 (2014)

[22] Ferreira, A.A., Veloso, A., Gonçalves, M.A., Laender, A.H.: Effective self-training author name disambiguation in scholarly digital libraries. In: Proceedings of the 10th Annual Joint Conference on Digital Libraries, pp. 39–48 (2010)

[23] Yang, K.-H., Wu, Y.-H.: Author name disambiguation in citations. In: 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, vol. 3, pp. 335–338 (2011). IEEE

[24] Arif, T., Ali, R., Asger, M.: Author name disambiguation using vector

space model and hybrid similarity measures. In: 2014 Seventh International Conference on Contemporary Computing (IC3), pp. 135–140 (2014). IEEE

[25] Qian, Y., Hu, Y., Cui, J., Zheng, Q., Nie, Z.: Combining machine learning and human judgment in author disambiguation. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 1241–1246 (2011)

[26] Sun, X., Kaur, J., Possamai, L., Menczer, F.: Detecting ambiguous author names in crowdsourced scholarly data. In: 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing, pp. 568–571 (2011). IEEE

[27] Hourrane, O., Mifrah, S., Bouhriz, N., Rachdi, M., *et al.*: Using deep learning word embeddings for citations similarity in academic papers. In: International Conference on Big Data, Cloud and Applications, pp. 185–196 (2018). Springer

[28] Ebraheem, M., Thirumuruganathan, S., Joty, S., Ouzzani, M., Tang, N.: Distributed representations of tuples for entity resolution. Proceedings of the VLDB Endowment **11**(11), 1454–1467 (2018)

[29] Hoffart, J., Yosef, M.A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., Weikum, G.: Robust disambiguation of named entities in text. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pp. 782–792 (2011)

[30] Han, X., Sun, L., Zhao, J.: Collective entity linking in web text: a graph-based method. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 765–774 (2011)

[31] Cao, K., Rei, M.: A joint model for word embedding and word morphology. arXiv preprint arXiv:1606.02601 (2016)

[32] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)

[33] Kuang, D., Ding, C., Park, H.: Symmetric nonnegative matrix factorization for graph clustering. In: Proceedings of the 2012 SIAM International Conference on Data Mining, pp. 106–117 (2012). SIAM

[34] Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: Online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 701–710

(2014)

[35] Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: Large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web, pp. 1067–1077 (2015)

[36] Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864 (2016)

[37] Tang, J., Qu, M., Mei, Q.: Pte: Predictive text embedding through large-scale heterogeneous text networks. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1165–1174 (2015)

[38] Hermansson, L., Kerola, T., Johansson, F., Jethava, V., Dubhashi, D.: Entity disambiguation in anonymized graphs using graph kernels. In: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, pp. 1037–1046 (2013)

[39] Baglioni, M., Manghi, P., Mannocci, A., Bardi, A.: We can make a better use of orcid: five observed misapplications. Data Science Journal **20**(1) (2021)